

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions and listings of claims in the application:

1. (Currently Amended) A method for defining an application for execution, wherein the application includes a graphical user interface and a functional component, the method comprising:

accessing a custom tag library, wherein the custom tag library includes one or more tags each designating an XPath operation ~~to access data~~;

defining a script to represent at least a portion of the graphical user interface; and

including in the defined script at least one of the tags to enable interaction between the graphical user interface and the functional component.

2. (Original) The method of claim 1, wherein the one or more tags in the custom tag library includes tags for performing one or more of the following functions: substituting attribute values, substituting element values, iteration, testing the validity of an expression, or creating a document object model instance.

3. (Original) A method for defining an application for execution, wherein the application includes a graphical user interface and a functional component, the method comprising:

accessing a custom tag library, wherein the custom tag library includes one or more tags each designating an XPath operation;

defining a JavaServer Page (JSP) to represent at least a portion of the graphical user interface; and

including in the defined JSP at least one of the tags to enable interaction between the graphical user interface and the functional component.

4. (Original) The method of claim 3, wherein the one or more tags in the custom tag library includes tags for performing one or more of the following functions: substituting attribute values, substituting element values, iteration, testing the validity of an expression, or creating a document object model instance.

5. (Original) A method for defining a user interface for interacting with a functional component of an application written in Java, the method comprising:

accessing a custom tag library, wherein the custom tag library includes one or more tags each designating an XPath operation; and

providing one or more instructions defining a user interface, wherein the one or more instructions provide for interaction between the user interface and the functional component,

wherein the one or more instructions do not include Java language.

6. (Original) A method for accessing data represented by a document object model (DOM) in a system including a custom tag library wherein each tag designates an XPath operation, the method comprising:

receiving an instruction to access data, wherein the instruction includes a tag associated with the custom tag library;

determining an action based on the included tag;

determining a target data on which the determined action is to be performed by parsing the instruction to obtain an XPath operation indicating the target data; and

performing the determined action on the target data.

7. (Currently Amended) A method for designing a user interface for an application, wherein the application includes data represented by a document object model, the method comprising:

receiving information about contents of the document object model; and

defining a user interface including one or more custom tags for accessing the contents of the document object model,

wherein the custom tags facilitate accessing the contents of the document object model by permitting use of XPath operations to access data.

8. (Original) A method for designing a user interface for an application, wherein the application includes data represented by a document object model, the method comprising:

receiving information about contents of the document object model; and

defining a user interface including one or more custom tags for accessing the contents of the document object model,

wherein the custom tags facilitate accessing the contents of the document object model by permitting use of XPath syntax.

9. (Currently Amended) The method of claim 8, wherein ~~the step of~~ defining a user interface includes ~~the step of~~:

defining a JavaServer Page (JSP) representing the user interface.

10. (Currently Amended) The method of 8, wherein ~~the step of~~ receiving information about the contents of the document object model includes ~~the step of~~:

receiving a document type definition.

11. (Original) A system for implementing an application including a user interface defined by a JavaServer Page (JSP), wherein the JSP includes custom tags each designating an XPath operation, the system comprising:

a JSP module for executing instructions in a user interface definition;

a custom tag library for resolving a custom tag included in the instructions in the user interface definition; and

a tag handler,

wherein the tag handler parses the custom tag included in the instructions and evaluates the XPath operation based on the custom tag.

12. (Currently Amended) A system for implementing an application including a user interface defined by a script, wherein the defined script includes custom tags each designating an XPath operation for accessing data, the system comprising:

a script module for executing instructions in a user interface definition;

a custom tag library for resolving a custom tag included in the instructions in the user interface definition; and

a tag handler,

wherein the tag handler parses the custom tag included in the instructions and evaluates the XPath operation for accessing data based on the custom tag.

13. (New) The method of claim 1, wherein a tag in the custom tag library provides a scope for an XPath expression.

14. (New) The method of claim 1, wherein a tag in the custom tag library allows iteration over an XPath expression.

15. (New) The method of claim 1, wherein a tag in the custom tag library obtains a value stored at a location represented by an XPath expression.

16. (New) The method of claim 1, wherein a tag in the custom tag library tests the validity of an XPath expression.

17. (New) The system of claim 12, wherein the custom tag provides a scope for an XPath expression.

18. (New) The system of claim 12, wherein the custom tag allows iteration over an XPath expression.

19. (New) The system of claim 12, wherein the custom tag obtains a value stored at a location represented by an XPath expression.

20. (New) The system of claim 12, wherein the custom tag tests the validity of an XPath expression.